Text Detection and Recognition in images: A survey

Tanvi Goswami
Dept. of Information Technology
Dharmsinh Desai University
Nadiad, India
tanvigoswami.it@ddu.ac.in

Zankhana Barad
Dept. of Information Technology
Dharmsinh Desai University
Nadiad, India
zankhanadabhi.it@ddu.ac.in

Prof. Nikita P. Desai
Dept. of Information Technology
Dharmsinh Desai University
Nadiad, India
npd_ddit@yahoo.com

Abstract- Text Detection and recognition is a one of the important aspect of image processing. This paper analyzes and compares the methods to handle this task. It summarizes the fundamental problems and enumerates factors that need consideration when addressing these problems. Existing techniques are categorized as either stepwise or integrated and sub-problems are highlighted including digit localization, verification, segmentation and recognition. Special issues associated with the enhancement of degraded text and the processing of video text and multi-oriented text are also addressed. The categories and sub-categories of text are illustrated, benchmark datasets are enumerated, and the performance of the most representative approaches is compared. This review also provides a fundamental comparison and analysis of the remaining problems in the field.

Keywords-Image processing, Text Detection, Text Recognition., Scene Text

I. INTRODUCTION

Detection of text and identification of characters in images is a challenging visual recognition problem. As in much of computer vision, the challenges posed by the complexity of these images and models that incorporate various pieces of high-level prior knowledge. In this paper, we list out results from a various methods that attempt to learn the necessary features directly from the data as an alternative to using purpose-built, text-specific features or models. In contrast to more classical OCR problems, where the characters are typically monotone on fixed backgrounds, character recognition in images is potentially far more complicated due to the many possible variations in background, lighting, texture and font. As a result, building complete systems for these scenarios requires us to invent representations that account for all of these types of variations. Indeed, significant effort has gone into creating such systems, with top performers integrating dozens of features and processing stages. Recent work in machine learning, however, has sought to create algorithms that can learn higher level representations of data automatically for many tasks. Such systems might be particularly valuable where specialized features are needed.

This paper is organized as follows. We will first survey some related work in scene text recognition, as well as the background and related work in Section II. We'll then describe the learning methodologies used in various experiments in Section III, and present various experimental results in Section IV followed by conclusions and references.

II. BACKGROUND

Graphic text and scene text are considered two basic classes of text, where the former refers to machine print text overlaid graphically and the latter refers to text on objects, captured in its native environment. Graphic text is usually machine printed, found in captions, subtitles and annotations in video and born-digital images on the web and in email. Scene Text, however, includes text on signs, packages and clothing in natural scenes, and is more likely to include handwritten material.

Scene text recognition as shown in fig.1 (b) and fig.3 has generated significant interest from many branches of research. While it is now possible to achieve extremely high performance on tasks such as digit recognition in controlled settings, the task of detecting and labeling characters in complex scenes remains an active research topic. However, many of the methods used for scene text detection and character recognition are predicated on cleverly engineered systems specific to the new task. For text detection, for instance, solutions have ranged from simple off-the-shelf classifiers trained on hand coded features to multi-stage pipelines combining many different algorithms.

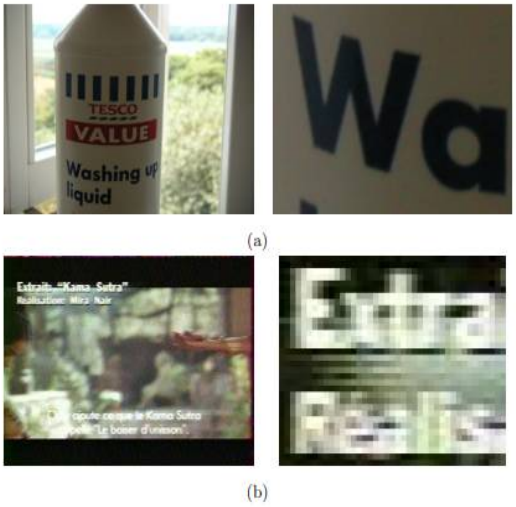
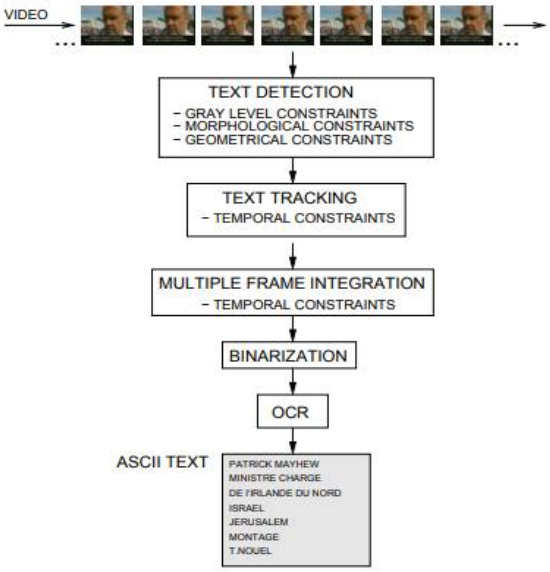


Figure 1: Example images and zooms into the text area: (a) artificial text; (b) scene text.

Figure 2: Caption Text Image (googlecode.blogspot.com)

Figure 3: Scene Text Image (ICDAR Dataset)



III. LEARNING METHODOLOGIES

3.1 Neural Network based model [1]:

The basic setup is closely related to a convolution neural network [2], but due to its training method it can be used to rapidly construct extremely large sets of features with minimal tuning. The system proceeds in several stages:

1) Apply an unsupervised feature learning algorithm to a set of image patches harvested from the training data to learn a bank of image features.
2) Evaluate the features convolutionally over the training images. Reduce the number of features using spatial pooling [15].
3) Train a linear classifier for either text detection or character recognition.

3.2 SVM based text detection in images [3]:

The below model's purpose is to develop a text model which takes into account the geometrical constraints directly in the detection phase: in first phase a first coarse detection calculates a text "probability" image. After wards, for each pixel they calculate geometrical properties of the eventual surrounding text rectangle. These features are added to the features of the first step and fed into a support vector machine classifier.

3.3 Stepwise Methodologies

Stepwise methodologies have four primary steps: localization, verification, segmentation, and recognition. The localization step coarsely classifies components and groups them into candidate text regions, which are further classified into text or non-text regions during verification. The segmentation step separates the characters so that exclusive, accurate outlines of image blocks remain for the recognition step. Finally, the recognition step converts image blocks into characters.

Text detection is performed with a convolution neural network [4] trained on raw pixel values, and the detected components of local maximal responses are grouped as text. A tracking process is integrated to determine the start and end frame of localized text. A segmentation step based on the Shortest Path method is proposed to calculate separations that enable accurate CNN based character recognition. A language model is then used to remove recognition ambiguities and segmentation errors.

Yao et al. [5], [6] developed an orientation robust, multilingual approach. Stroke pixels were grouped into connected components (CCs), which were filtered with a decision forest trained on component features of shape, occupation ratio, axial ratio, width variation, and component density. Filtered connected components were then aggregated into multi-oriented chains with a hierarchical clustering algorithm, and verified by a decision forest classifier trained on region features including color, density, stroke, and structure. The chains that pass verification are enhanced by a low rank structure recovery algorithm, and are then fed to an OCR module to produce recognition results.

3.4 Integrated Methodologies

With an integrated methodology, character classification responses are considered the primary cues, and shared with detection and recognition modules [7]. K.Wang et al use

nearest neighbor classifier to do the multi-scale sliding window classification to obtain character responses, and the non-maximum suppression to localize character candidates. They employ the pictorial model that takes the scores and locations of characters as input to determine an optimal configuration of a particular word from a small lexicon. T.Wang et al. [9] proposed combining a multi-layer CNN with unsupervised feature learning to train character models, which are used in both text detection and recognition procedures. As shown in Fig. below:

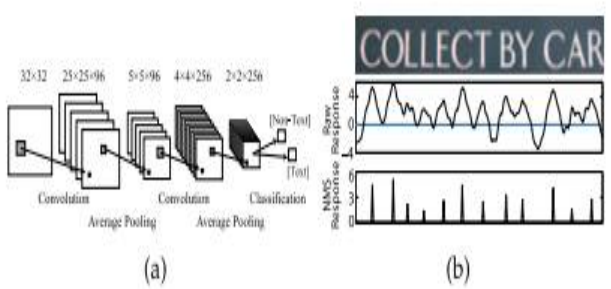


Fig: CNN based integrated detection & recognition approach: (a) CNN for character detection (b) CNN responses for recognition

They run CNN based sliding window character classification and use the responses to localize candidate text lines. They then integrate the character responses with character spacing and a define lexicon using a beam search algorithm [10] to recognize words. A dynamic programming algorithm is used to select the path on the graph with the highest score. The sequence of regions and their labels induced by the optimal path are the outputs, i.e., a word, a sequence of words or a non-text region.

IV. COMPARISON OF EXPERIMENTAL RESULTS

The result generated by Adam coates et al.[1] using unsupervised feature learning on scene images is 85.5% on sample size 36, Wang et al.[9] has used HOG and Random Ferns based character model, pictorial model optimization techniques for small size lexicons. They have used ICDAR'03 dataset and SVT protocol. They found 50/1,156 lexicons. Authors have achieved Word Recognition Accuracy 0.760/0.620 and highlights word spotting. Wang et al.[8] has used CNN based character modeling, Beam search based optimization with a lexicon . They have used ICDAR'03 dataset and SVT protocol. They found 50/1,156 lexicons. Authors have achieved Word Recognition Accuracy 0.900/0.840.

V. CONCLUSION

The technology of text detection and recognition has grown vastly over the last two decades and many real time vision systems take advantage from it. We have discussed briefly some of the different methods for text detection and processing text in images. Among these the CNN based technique proposed by Wang et al.[8] gives promising results and claims an accuracy of about 90% in their experiments.

Automation of tedious tasks such as post sorting, forms processing, video surveillance reduces the time taken to complete the task significantly and makes life a lot easier. For example Google Cloud Vision API[11] enables developers to understand the content of an image by encapsulating powerful machine learning models in an easy to use REST API. It quickly classifies images into thousands of categories (e.g., "boat", "lion", "monuments"), detects individual objects and faces within images and finds and reads printed words contained within images. We can build metadata on your image catalog, moderate offensive content, or enable new marketing scenarios through image sentiment analysis. Analyze images uploaded in the request or integrate with your image storage on Google Cloud Storage.

REFERENCES

[1] Adam Coates, Blake Carpenter, Carl Case, Sanjeev Satheesh, Bipin Suresh, Tao Wang, David J. Wu, Andrew Y. Ng, "Text Detection and Character Recognition in Scene Images with Unsupervised Feature Learning"

[2] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," Neural Computation, vol. 1, pp. 541–551, 1989

[3] Christian Wolf ,Jean-Michel Jolion, "Model based text detection in images and videos: a learning approach"

[4] M. Delakis and C. Garcia, "Text detection with convolutional neural networks," in Proc. Int. Conf. Comput. Vis. Theory Appl., 2008, vol. 2, pp. 290–294.

[5] C. Yao, X. Zhang, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., 2012, pp. 1083–1090.

[6] C. Yao, X. Zhang, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Rotationinvariant features for multi-oriented text detection in natural images," PLoS ONE, vol. 8, no. 8, p. e70173, 2013.

[7] W. Wu, D. Chen, and J. Yang, "Integrating co-training and recognition for text detection," in Proc. IEEE Int. Conf. Multimedia Expo, 2005, pp. 1166–1169.

[8] K. Wang and S. Belongie, "Word spotting in the wild," in Proc. Eur. Conf. Comput. Vis., 2010, pp. 591–604.

[9] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolution neural networks," in Proc. IEEE Int. Conf. Pattern Recognit., 2012, pp. 3304–3308.

[10] C. L. Liu, M. Koga, and H. Fujisawa, "Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 11, pp. 1425–1437, Nov. 2002.

[11] https://cloud.google.com/vision/